

## ТЕСТУВАННЯ ANDROID ДОДАТКІВ, НАПИСАНИХ З ВИКОРИСТАННЯМ ПАТЕРНІВ MVC, MVP ТА MVVM

*магістр Д.В. Никифоров, канд. фіз.-мат. наук, доц. О.П. Черних,  
Національний технічний університет "Харківський політехнічний  
інститут", м. Харків*

Для динамічного розвитку Android необхідно ефективно тестування мобільних додатків, враховуючи нові версії операційної системи. Автоматизоване функціональне тестування допомагає забезпечити необхідну функціональність мобільного застосування та належну роботу інтерфейсу користувача. Однак існує необхідність вийти за рамки функціонального тестування мобільних додатків. Існує безліч аспектів продуктивності мобільних додатків, які потрібно вимірювати і покращувати. Є прямий взаємозв'язок між продуктивністю і зручністю використання мобільних додатків.

Тестування є важливою частиною процесу розробки мобільних додатків. Для Android воно має особливу важливість, оскільки використовувані пристрої досить сильно відрізняються один від одного за такими параметрами: розміром і дозволом екрану; версією Android; форм-фактором пристрою; системою команд процесора; фронтальною камерою, NFC, зовнішньою клавіатурою і т.д.

Процес тестування додатків може включати тести різних типів. Можна регулярно проводити автоматичне тестування функціональності без додаткових витрат. Наприклад, можна на ніч запускати тестування збірки на всіх пристроях, а вранці аналізувати результати і виправляти помилки.

Тестування додатків, написаних з використанням різних архітектурних патернів, відрізняється тільки у плані написання юніт тестів. Для цих потреб використовується фреймворк `jUnit`.

Важливою частиною підготування коду до тестування є виділення компонентів без використання класів з Android фреймворку. Це дозволить запускати тести на локальний JVM, що, в свою чергу, зробить процес тестування набагато швидшим. У архітектурі MVC таким компонентом має бути Контролер, у MVP – Презентер, у MVVM – Модель представлення.

Для тестування логіки інтерфейсу користувача використовується бібліотека Espresso, яка дозволяє виконувати запис послідовності дій, перевіряти кінцевий результат та стан інтерфейсу на момент закінчення тестування. Дане тестування не зв'язане з архітектурою додатка.